

## Sprint 2 Report

**Team:** eLation Nation

**Project:** eLation Mobile Apps

**Project Sponsor:** Innovative Systems LLC

**Sprint Duration:** October 10, 2012 to October 31, 2012

### Team members:

- Michael Malkowski
- Rachel Pekarek
- Jeremy Warner

**Project Sponsor:** Innovative Systems LLC

### Meetings/Client Interactions

- 10/11/12
  - Scrum Meeting
    - Updated Client on status
- 10/16/12
  - Scrum Meeting
    - Updated Client on status
- 10/18/12
  - Scrum Meeting
    - Met with Dave and Brian to determine how we will handle persistent data storage in the app
- 10/23/12
  - Had first phone conference with Chris, our coworker in Mitchell, who is helping us with our API
  - Scrum Meeting
    - Updated Client on status
- 10/25/12
  - Scrum Meeting
    - Updated Client on status
- 10/30/12
  - Scrum Meeting
    - Updated Client on status

## **Sprint Tasks**

Our sprint tasks were divided into the following categories:

- Android
  - Investigate PDF for Android – {Mike}
  - Get Main UI set up – {Mike}
  - Implement Core UI – {Mike}
- Apple
  - Investigate PDF for iOS – {Rachel}
  - Get Main UI set up – {Rachel}
- API
  - Research common code for iOS and Android to do JSON calls – {Jeremy}
  - Build Android data object – {Jeremy}
  - Build individual GET data functions Android – {Jeremy}
  - Build iOS data object – {Rachel, Jeremy}
  - Add error handling to Apple APICall object – {Rachel, Jeremy}
  - Build Individual GET data functions iOS – {Jeremy}
  - Current Usage Data for Android – {Jeremy}
- Documentation
  - Power Point Presentation – {Mike, Rachel, Jeremy}
  - UI Design Master Document – {Mike, Rachel}
  - Sprint 2 Report – {Mike, Rachel, Jeremy}

## Goals Met

This section will detail the goals we were able to meet for this sprint.

- Android
  - Investigate PDF for Android – {Mike}
    - Defined our solution for this platform, but we were unable to get a PDF from the API to display.
  - Get Main UI set up – {Mike}
    - Finishing the UI skeleton
  - Implement Core UI – {Mike}
    - Partially completed. We got some data back from the API calls and into the display, but not as much as this task detailed. We should not have listed the entire set of functionality to be completed as we did.
    - This was also limited by how fast Jeremy was able to get the API calls done and working.
- Apple
  - Investigate PDF for iOS – {Rachel}
    - Defined our solution for this platform, but we were unable to get a PDF from the API to display.
  - Get Main UI set up – {Rachel}
    - Finishing the UI skeleton
- API
  - Research common code for iOS and Android to do JSON calls – {Jeremy}
    - JSON parsing libraries exist in both Android and iOS 5 environments. Building a low level common parsing library (C/C++) would save time if the API was extensive, but due to the current size and the overhead involved with porting libraries into iOS and Android we have opted not to build a common parsing library at this time.
    - This may be revisited at a later date if needed.
  - Build Android data object – {Jeremy}
    - Object built. Needs more error handling done.
  - Build individual GET data functions Android – {Jeremy}
    - All done except for Current Usage call.
  - Build iOS data object – {Rachel, Jeremy}
    - Done. Needs review to ensure proper design
  - Add error handling to Apple APICall object – {Rachel, Jeremy}
    - Error Handling has been added.
  - Build Individual GET data functions iOS – {Jeremy}
    - Partially completed (6/10). As a stretch goal, we were unsure how long the iOS API interface was going to take. Jeremy is happy with how much he got done.
- Documentation
  - Power Point Presentation – {Mike, Rachel, Jeremy}
    - Done and presented
  - UI Design Master Document – {Mike, Rachel}
    - Started (Living document)
  - Sprint 2 Report – {Mike, Rachel, Jeremy}
    - Done

## Goals Not Met

This section will detail the goals we did not meet for this sprint

- Android
  - Implement Core UI – {Mike}
- API
  - Build Individual GET data functions iOS – {Jeremy}
  - Current Usage Data for Android – {Jeremy}
    - This API call turned out to return more data than originally thought, delaying implementation. In addition, it was not working until 10/29/2012. We did not discover that it was not functioning correctly until the week of 10/22/2012. We could have checked it sooner if we had tested the API calls sooner. This will be fixed during the next sprint.
- Documentation
  - UI Design Master Document – {Mike, Rachel}
    - We would have liked to have done more on this document. Unfortunately we were unable to find the time to work on this document as it was of a lower priority. It will be addressed further next sprint.

## Research/Code Experiments

- Mike and Rachel researched proper ways of viewing PDFs in Android and iOS
- Jeremy learned about writing in Objective-C

## Prototypes & Features

- Functional UI applications for both Android and iOS
- Real data is implemented in Android's Current Invoice Summary

## Product Backlog

- Sign in pages using OpenAuth on both platforms
- Implement PDF viewer and storage on both platforms
- Set up Databases (Persistent data)
- Build POST data functions and methods into API calls
- Implement the Log Pages
- CONOPS, Requirements, and Design Documents
- Use API calls to fill UI with real data
- Determine how to handle Previous Invoice and Payment History Pages
- Make iOS app compatible with the iPad
- Check password strengths and show a progress bar
- Valid email verification
- Investigate Trouble Tickets
- Build Unit Tests
- Set up API and UI for Technician App